



HoGent

HoGent

BEDRIJF
EN
ORGANISATIE

DATABANKEN II XML

XML DOCUMENT

XML-DOCUMENT

Hoofdstuk 1 : XML

- Wat is XML?
- Structuur van een XML document
 - Prolog
 - De xml declaratie
 - overige
 - Epilog
 - Body
 - Elementen
 - Attributen
 - Entity References
 - Well-formed XML
 - Boomstructuur
- Opbouwen van XML document
- Het leven van een XML document
 - CDATA secties
 - Commentaar
 - Witruimtes

Hoofdstuk 1 : XML

Voorbeeld HTML

<p>

Mrs. Mary McGoon

1401 Main Street

Anytown, NC 34829

</p>

Hoofdstuk 1 : XML

Voorbeeld XML

<adres>

Mrs. Mary McGoon

1401 Main Street

Anytown, NC 34829

</adres>

Hoofdstuk 1 : XML

Voorbeeld XML

```
<adres>  
  <naam>  
    <aanspreking>Mrs. </aanspreking>  
    <voornaam>Mary</voornaam>  
    <familienaam> McGoon</familienaam>  
  </naam>  
  <straatenummer>1401 Main Street</straatenummer>  
  <stad>Anytown</stad>  
  <state>NC </state>  
  <postcode>34829</postcode>  
</adres>
```

1. Wat is XML?

- XML : eXtensible Markup Language.

Markup: XML is een opmaaktaal. Gegevens worden gestructureerd omsloten door tags. De tags definiëren en beschrijven elke vorm van informatie. XML geeft betekenis aan data.

Extensible: XML kent geen voorgedefinieerde tags en geen voorgedefinieerde opmaak, en legt ook geen beperkingen op.

De gebruiker definieert zelf de structuur en semantiek van zijn document adhv **Markup**

1. Wat is XML?

- Een beetje historiek
 - SGML (Standard Generalized Markup Language)
 - Een ISO standaard uit 1986
 - **Is een meta-markup-taal**; D.i. een taal waarin men zijn eigen markup talen voor documenten kan definiëren, door definitie van een tag set en van de structurele verbanden binnen de tags.
 - Grote nadeel : complexiteit
 - Meest gekende toepassing van SGML is HTML. De tags van HTML (tonen en linken van informatie) zijn vastgelegd m.b.v. SGML.

1. Wat is XML?

- Een beetje historiek
 - HTML is een standaard voor layout van webdocumenten
 - Is een SGML applicatie
 - Gericht op ‘menselijke’ verwerking MAAR veel minder geschikt voor ‘computer’-verwerking.
 - Beperkingen :
 - Uitbreidbaarheid
 - Mogelijkheid om structuur aan te brengen in documenten
 - Gegevens in documenten kunnen niet gevalideerd worden

1. Wat is XML?

- Een beetje historiek
 - XML
 - Standaard (XML 1.0) gedefinieerd door W3C sinds februari 1998 <http://www.w3.org/TR/xml>
 - « **SGML voor het web** » : subset van SGML, maar veel eenvoudiger
 - doel : vereenvoudigen van de overdracht van gestructureerde informatie over het internet
 - XML is ook **een meta-markup-taal** : een taal waarin men zijn eigen talen kan ontwikkelen, zoals XHTML
 - XML is een verzameling regels: deze schrijven voor hoe tags eruit zien en hoe ze mogen gebruikt worden om tekstfragmenten in een tekstbestand te markeren. De namen van de tags zijn niet vastgelegd.

1. Wat is XML?

- Een beetje historiek
 - XML
 - Versies :
 - XML 1.0 is gebaseerd op Unicode 2.0 :
 - » XML 1.0 noemt expliciet alle tekens die in elementnamen, attribuutnamen,... mogen worden gebruikt. Tekens niet gedefinieerd in UniCode 2.0 mogen niet gebruikt worden voor markup in XML 1.0, wel in de tekst.
 - XML 1.1 daarentegen is onafhankelijk van elke Unicode versie
 - Richtlijn : gebruik versie 1.0.
 - XHTML
 - een applicatie van XML
 - Belangrijkste verschillen met HTML
 - XHTML elementen moeten correct genest zijn
 - XHTML elementen moeten altijd gesloten zijn
 - XHTML elementen moeten in kleine letters staan
 - XHTML documenten moeten een root element hebben

1. Wat is XML?

- Waarom zo'n hype rond XML????
 - Geen speciale editors nodig
 - Geen specifieke technologie nodig
 -
- Wat maakt het dan zo speciaal?

1. Wat is XML?

- XML is een meta-taal: taal die een andere taal beschrijft
- XML is volledig uitbreidbaar
- XML is gestructureerd
 - expliciet : tags en elementen
 - impliciet : hiërarchische structuur
- XML wordt structureel gevalideerd
- XML biedt meerwaarde voor zoekmachines
- XML is tekst-gebaseerd en makkelijk om te lezen =>
- XML is platformonafhankelijk => XML documenten zijn makkelijk uitwisselbaar
- XML is een open standaard : <http://www.w3.org/>
- XML is (programmeer-) taalonafhankelijk
- XML is media-neutraal = scheidt presentatie en inhoud
- XML is eenvoudig te updaten

1. Wat is XML?

- XML is ideaal als **uitwisselingsformaat**
 - Unicode, platform-onafhankelijk, fabrikant-, hardware-, software-, protocol-onafhankelijk, uitbreidbaar
 - De structuur van een XML document kan je vastleggen
 - => je kan de geldigheid van een document bepalen
 - DTD (Document Type Definition) (W3C standaard)
 - Structuur: welke elementen en attributen, volgorde, frequentie
 - Maar : beperkte gegevenstypes; GEEN XML syntax
 - XML Schema (W3C standaard)
 - Uitgebreidere beschrijving structuur en datatypes mogelijk
 - XML syntax
 - RelaxNG (ISO standaard)
 - Eenvoudiger dan schema. Minder beperkingen
 - Maar wordt zelden gebruikt
 - Schematron (ISO standaard)
 - Definiëren van regels

- Voorbeeld DTD

The image shows two side-by-side code editors. The left editor, titled 'memo.xml', contains an XML document with a root element 'memo' containing several child elements: 'aan' (with values 'Jan' and 'Karel'), 'van' (with value 'Piet'), 'kop' (with value 'Let op'), and 'tekst' (with value 'dit is een voorbeeld met een externe dtd'). The right editor, titled 'memo.dtd', contains a DTD definition for the 'memo' element and its children. The DTD defines 'memo' as a container for 'aan', 'van', 'kop', and 'tekst' elements, all of which are defined as text elements (#PCDATA).

```
memo.xml
1 <?xml version="1.0" encoding="UTF-8"?>
2 <!DOCTYPE memo SYSTEM "file:/E:/XML/memo.dtd">
3 <memo>
4   <aan>Jan</aan>
5   <aan>Karel</aan>
6   <van>Piet</van>
7   <kop>Let op</kop>
8   <tekst>dit is een voorbeeld met een externe dtd</tekst>
9 </memo>
```

```
memo.dtd
1 <?xml version="1.0" encoding="UTF-8"?>
2 <!--DTD voor memo-->
3 <!ELEMENT memo (aan+, van, kop,tekst)>
4 <!ELEMENT aan (#PCDATA)>
5 <!ELEMENT van (#PCDATA)>
6 <!ELEMENT kop (#PCDATA)>
7 <!ELEMENT tekst (#PCDATA)>
```


1. Wat is XML?

- XML is ideaal als **uitwisselingsformaat**
 - XML applicaties : domein-specifieke talen gebaseerd op XML
 - W3C
 - XHTML
 - MathML : Wiskunde (www.w3.org/Math)
 - SMIL : Synchronized Multimedia Integration Language (www.w3.org/TR/smil20/)
 - SVG : Vector graphics (www.w3.org/TR/SVG/)
 - SOAP (www.w3.org/TR/soap/)
 - ISO
 - OOXML
 - Industrie – specifieke talen
 - Human Resource : <http://www.hr-xml.org>
 - ...
Overzicht op <http://xml.coverpages.org/xmlApplications.html>

1. Wat is XML?

- Bij XML is **inhoud gescheiden van opmaak**
 - XML vormgeven via **XSL / CSS**
 - <http://www.w3schools.com/xsl/tryxslt.asp?xmlfile=cdcatalog&xsltfile=cdcatalog>
 - 2 belangrijke toepassingen
 - Verwerking door computers : een XML document wordt getransformeerd naar een andere XML vorm, aangepast aan applicatie die document zal verwerken.
 - Verwerking door een menselijke eindgebruiker : vanuit één XML document output mogelijk naar zowel papier(pdf), web (html), visualisatie op mobiele telefoon (WML),... en per medium zijn meerdere *dataviews* mogelijk
 - Voordelen
 - Single Source : 1 bron, meerdere visualisaties
 - Consistentie van gegevens : gegevens dienen maar op 1 plaats te worden bijgehouden
 - Verschillende gelijkaardig gestructureerde documenten kunnen op een éénvormige wijze gevisualiseerd worden met dezelfde stylesheet

1. Wat is XML?

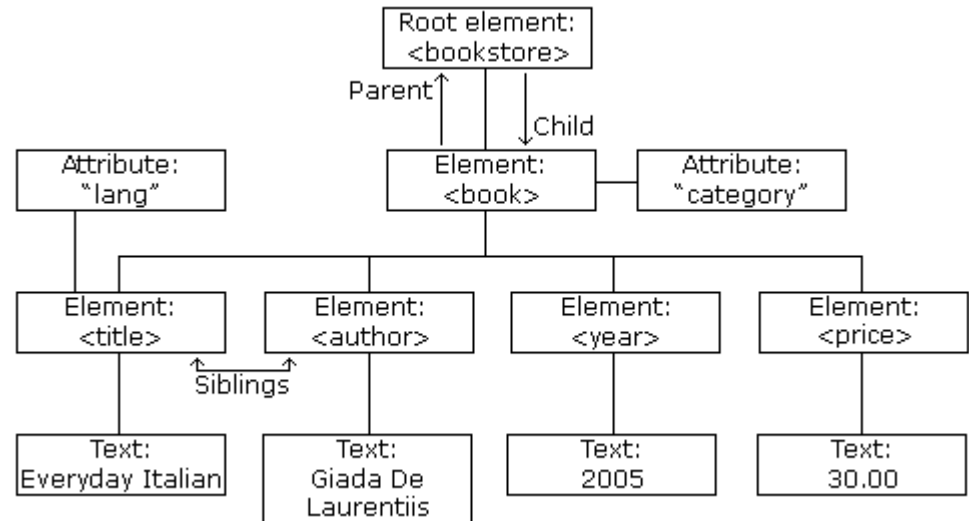
- Verwerking van XML-documenten in een applicatie (Java, .NET,...) mogelijk via:

1) DOM (Document Object Model)

= Tree gebaseerde API, die boomvoorstelling van document bouwt in het geheugen. Biedt mogelijkheden voor navigeren en manipuleren van XML documenten.

2) SAX (Simple API for XML)

= Event gebaseerde API. Het document wordt als stream verwerkt. Enkel voor lezen van XML documenten.



1. Wat is XML?

- Efficiënt zoeken in XML document
 - De zoekmethode is afhankelijk van de vorm waarin een document is opgeslagen, en het soort van meta-informatie dat beschikbaar is
 - Soorten zoekopdrachten
 - Full text search
 - Vb: Zoek de beursnotering van Dell : “beursnotering Dell”
 - Keyword-based search : o.b.v. Document metadata
 - Vb: Zoek de beursnotering van Dell maar enkel in documenten van de afgelopen 6 maanden
 - Structured search : o.b.v. Structural metadata.
 - Zoek in documenten van het type beursnotering naar `<company>Dell</company>`
→ veel efficiënter
 - XQuery : query taal voor XML
 - Semantic Web : zoeken op basis van semantiek

1. Wat is XML?

- XML bevordert de ontwikkeling van nieuwe technieken en XML technologieën
 - Webservices : WSDL, SOAP, UDDI
 - RSS-feeds
 - Ajax
 - WAP en WML (Wireless Markup Language)
 - XML Databases
 - XML Digital Signature (w3.org/TR/xmlsig-core/)
 - XML Encryption (w3.org/TR/xmlenc-core/)
 - XAML (.NET 3.0 : WPF)
 - ODF (Open Document Format) <-> OOXML(Office Open XML)
 - ...

1. Wat is XML?

– RSS (Really Simple Syndication)

```
<?xml version="1.0" encoding="UTF-8" ?>
<rss version="2.0"><channel>
  <title>W3Schools Home Page</title>
  <link>http://www.w3schools.com</link>
  <description>Free web building tutorials</description>
  <item>
    <title>RSS Tutorial</title>
    <link>http://www.w3schools.com/rss</link>
    <description>New RSS tutorial on W3Schools</description>
  </item>
</channel>
</rss>
```

1. Wat is XML?

- Ajax (Asynchroon JavaScript en XML)
 - Gevraagde gegevens worden asynchroon opgehaald van de webserver
 - => De pagina's hoeven niet in het geheel ververst worden
 - Voorbeeld
Google



xml			Geavanceerd zoeken
	Google zoeken		Voorkeuren
xml		395.000.000 resultaten	Taalhulpmiddelen
xmling		161.000 resultaten	
xml editor		17.000.000 resultaten	
xmind		126.000.000 resultaten	
xmbo		1.150.000 resultaten	
xmediatrade		27.200 resultaten	
xmlspy		989.000 resultaten	
Ad xmlhttprequest		1.380.000 resultaten	ish
xml notepad		378.000 resultaten	
xml schema		3.740.000 resultaten	
			sluiten

1. Wat is XML?

– Ajax - Voorbeeld

- De HTML form

```
form> First Name:  
<input type="text" id="txt1" onkeyup="showHint(this.value)">  
</form>  
<p>Suggestions: <span id="txtHint"></span></p>
```

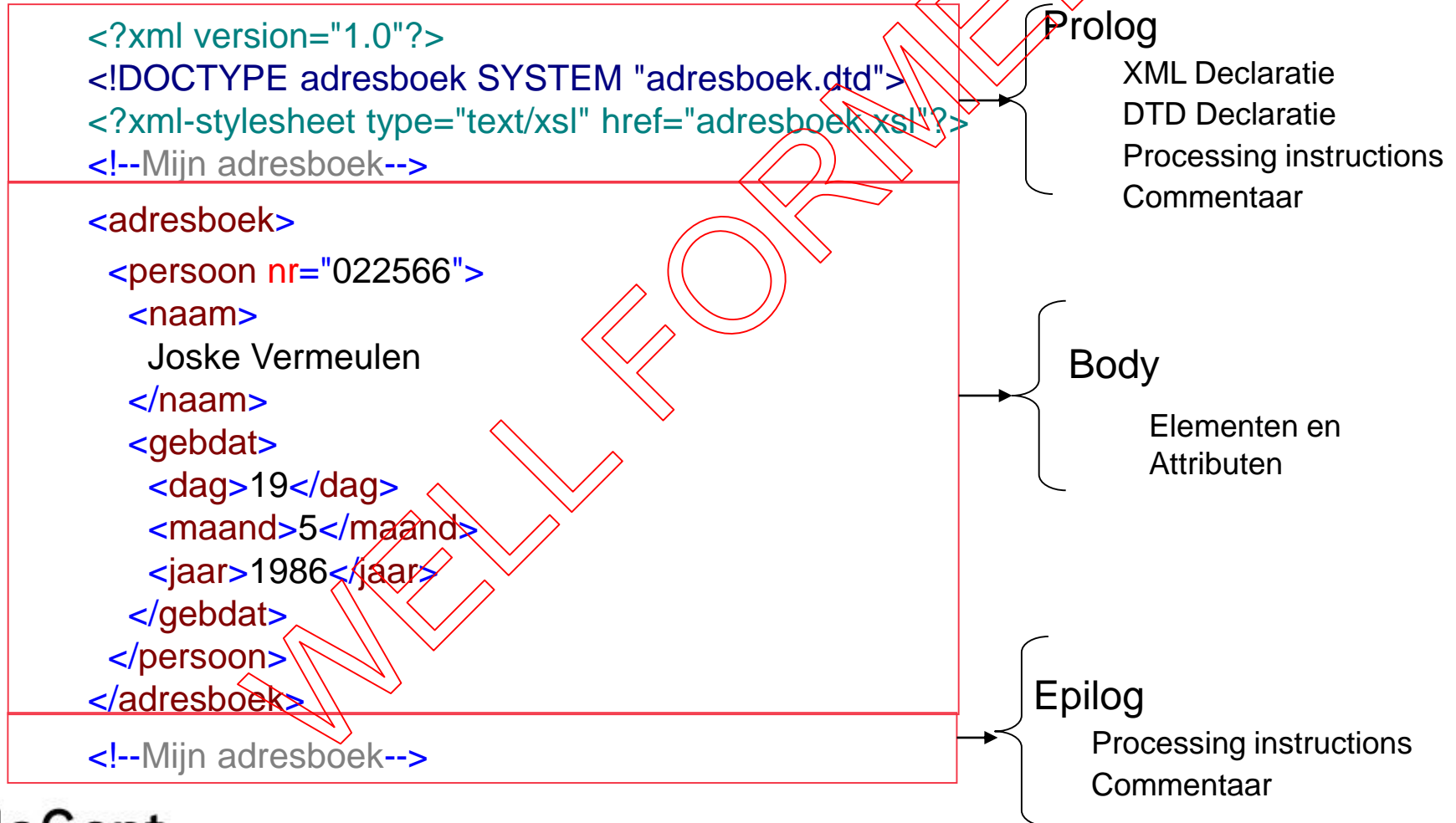
Type a Name in the Box Below

First Name:

Suggestions: Anna , Amanda

2. Structuur XML document

- Voorbeeld XML document (adresboek.xml)
 - Meestal opgeslaan in een tekstbestand met extensie .xml



2. Structuur XML document

a. Prolog (header)

- Begin van een XML document. Bevat :
 - De XML declaratie
 - `<?xml version="1.0" encoding="UTF-8"?>`
 - Version : versie van de XML specificatie. Gebruik 1.0 (vermijd 1.1)
 - Encoding (optioneel) : character encoding van document
 - » UTF-8 : meest economisch: gebruikt 1 byte voor ASCII-set (default)
 - » UTF-16 : 2 bytes per character
 - » ISO-8859-1 : bevat alle characters gebruikt in West Europese talen
 - » K016-R : Russische encoding
 - De Document Type declaratie
 - Andere processing instructies (PI) `<?targetapplicatie ...?>`
 - Bevatten info voor de parser of andere applicaties.
 - Bvb koppelen van een stylesheet aan een XML document
 - XML Commentaar `<!--.....-->`

2. Structuur XML document

b. De epilog (optioneel)

- Einde van XML document. Bevat :
 - Processing instructies (PI)
 - XML Commentaar <!--.....-->

2. Structuur XML document

c. De body

- De body bestaat uit
 - Elementen
 - Attributen
 - Entity references
 - CDATA sectie
 - Commentaar
 - Witruimte

2. Structuur XML document

c.1. Elementen

- Bestaan uit een combinatie van een start-tag, (optionele) inhoud en een eind-tag
- Tags
 - Vormen de markup van het document
 - Beschrijven de inhoud die ze omsluiten
 - Bestaat uit een <, een tagnaam en >
 - Elke starttag moet worden afgesloten
 - » Eindtag : ``
 - » Of verkort : `` (= leeg element)
 - Tagnamen zijn hoofdlettergevoelig (de open tag moet gelijk zijn aan de sluit tag).
 - Tagnamen zijn volledig vrij. XML is extensible (<->(X)HTML)
 - Tags mogen niet overlappen

2. Structuur XML document

c.1. Elementen

- Tags
 - Regels voor naamgeving van tags (en attributen)
 - » Een tag moet starten met een letter of een underscore.(Mag niet beginnen met een nummer)
 - » Kan zowel letters als nummers bevatten
 - » Spaties zijn niet toegelaten
 - » Mag niet beginnen met xml, d.i. een gereserveerd woord.
 - » Gebruik liever geen “:” in naam (namespaces)
 - » Gebruik liever geen – of . maar de “camelCasing” notatie
`<docentNaam>` ipv `<docent_naam>`
 - » Gebruik betekenisvolle namen

2. Structuur XML document

c.1. Elementen

- Inhoud: 4 mogelijkheden:
 - Enkel tekst => PCDATA (Parsed Character Data : geen <, >, &, “, ‘)
`<student>Harry Potter</student>`
 - Container voor andere elementen
 - » MAAR elementen mogen niet overlappen (volgorde is belangrijk)
`<student>`
`<voornaam>Harry</voornaam>`
`<naam>Potter</naam>`
`</student>`
 - Gemengd
`<student>Harry Potter<vak>XML</vak></student>`
 - Empty (leeg element)
 - » `
</br>` of verkort `
`
- Elementen zijn uitbreidbaar (xml is extensible)
- Elk element heeft 0,1 of meerdere attributen

2. Structuur XML document

c.2. Attributen

- Extra informatie binnen een tag. Hierdoor kan je een tag bepaalde eigenschappen meegeven.
- Staat in de start tag
- Heeft een naam en een waarde (van het type CDATA)
- Attribuutwaarde wordt altijd omsloten door quotes
 - ‘single quotes’ of “double quotes” toegelaten
- Elk element heeft 0, 1 of meerdere attributen
 - maar de attributen mogen niet dezelfde naam hebben
 - Volgorde attributen is onbelangrijk
 - » Uitzondering: XML header
- Regels naamgeving attributen : idem tags

```
<student studentnr="1232">Harry Potter</student>  
<img src = 'img.gif'/>
```


2. Structuur XML document

c.2. Attributen

- Attributen <-> Elementen?

```
<person sex="female">  
  <firstname>Anna</firstname>  
  <lastname>Smith</lastname>  
</person>
```

```
<person>  
  <sex>female</sex>  
  <firstname>Anna</firstname>  
  <lastname>Smith</lastname>  
</person>
```

- Geen officiële regels
- Algemene consensus
 - » Elementen voor data
 - » Attributen voor informatie over data (meta-data)
- Redenen om geen attributen te gebruiken
 - » Attributen mogen geen meervoudige waarden bevatten
 - » Attributen kunnen geen substructuren bevatten
 - » Attributen zijn niet geordend, elementen wel
 - » Moeilijker te manipuleren in programma's

2. Structuur XML document

c.3. Entity References (entiteitsverwijzingen)

- Kan je pas gebruiken als vooraf gedefinieerd in DTD
- Markup die de parser vervangt door gerefereerd object
 - Afkorten van lange of moeilijke teksten, die vaak terugkomen
 - Vastleggen van tekens, die niet met een karakter kunnen weergegeven worden door het computersysteem
 - Een verwijzing naar een extern bestand
- Begint met een & en eindigt met een ;
- 5 voorgedefinieerde entiteiten in XML:
 - » < (<) en > (>)
 - » & (&)
 - » " (") en ' (')

2. Structuur XML document

c.3. Entity References (entiteitsverwijzingen)

- Voorbeeld van andere entiteiten. Definitie staat in DTD!!
 - `<!ENTITY copyright "copyright Hogeschool Gent">`
 - » Gebruik : `<footer>©right;</footer>`
 - `<!ENTITY euro "€">` : unicode voor Euro teken
 - `<!ENTITY test SYSTEM "xml.txt">`
 - » Gebruik : `<hello>Hello &test;</hello>`

2. Structuur XML document

c.4. CDATA secties

- Inhoud van een CDATA-stuk wordt NIET behandeld bij het parsen van een XML-bestand. De formattering blijft behouden.
- In een CDATA-stuk kan je dus gebruik maken van `<`, `>`, `&`, `"`, `'`
- Je kan niet gebruik maken van `[]`
- Algemeen
`<![CDATA[Ignore this data please.]]>`
- Voorbeeld

`<vergelijking><![CDATA[6 < 7 & 7 > 6]]></vergelijking>`

2. Structuur XML document

c.5. Commentaar

- Start tag (<!--)
- Eind tag (-->)
- <!-- Dit is commentaar -->
- Regels voor commentaar
 - Geen spaties in begin en eindtag
 - Mag niet in de tag zelf
 - <el <!--Dit mag niet -->>
 - Alles tussen commentaar wordt genegeerd door de XML parser. Een verwerkend programma kan wel aan het commentaar.

2. Structuur XML document

c.6. Witruimtes

- Een XML parser moet alle karakters tussen tags doorgeven aan de applicatie. Dus ook alle witruimtes.
 - In voorbeeld hebben beide een verschillende inhoud
 - » `<naam>Jan Janssens</naam>`
 - » `<naam>Jan Janssens</naam>`
- Een LF/CR of combinatie wordt herleid tot een single LF om lijnen te separeren
- Maar het is aan de applicatie om de white space verder af te handelen.
- Opmerking :
 - Indenteren in XML bestand, maakt XML bestand leesbaarder voor de mens maar groter!!
 - Als geen indentering kan je xml bestand evengoed bekijken in IE, ... en dan wordt de structuur wel duidelijk.

2. Structuur XML document

d. Well-formed XML

- Een XML-document moet voldoen aan de syntax regels beschreven in de XML 1.0 Specification
<http://w3.org/TR/REC-xml>
 - Een XML-document moet altijd 1 uniek root element hebben
 - Elk element heeft een openingstag en een sluittag
 - Elementen moeten correct genest zijn
 - De tags zijn hoofdlettergevoelig
 - Attribuutwaarden moeten tussen aanhalingstekens staan
 - De tekens < en & moet correct gecodeerd zijn nl. via de entiteitsnamen < en &

2. Structuur XML document

- Oefeningen :
 - XML well-formed?

```
<?xml version="1.0"?>
<persoon nr="022566">
  <naam>
    Joske Vermeulen
  <gebdat>
    <dag>19
    <maand>5
    </dag>
    </maand>
    <Jaar>1986</jaar>
  </gebdat>
</persoon>
<persoon nr="022567">
  <naam>
    Joske D'Haenen
  </naam>
</Persoon>
```


2. Structuur XML document

- Oefeningen
 - Welke combinaties zijn juist?
 - A: <Boek id=T12 taal=NL ISBN=000-123-56/>
 - B: <Boek id='T12' taal='NL' ISBN='000-123-56'/>
 - C: <Boek id="T12" taal="NL" ISBN="000-123-56"/>
 - D: <Boek taal="NL" taal="E" ISBN="000-123-56"/>

2. Structuur XML document

- Oefeningen

- Zijn dit well-formed XML documenten?

```
<?xml version="1.0" standalone="yes" ?>
```

```
<Boek>
```

```
<TITEL>Professional XML</Titel>
```

```
<Para>Why XML?</para>
```

```
</Boek>
```

```
<?xml version="1.0" standalone="yes" ?>
```

```
<Boek>
```

```
<Para>
```

```
<bold>Bold text <italic>bold en italic stukje tekst</bold>italic stukje  
tekst</italic>
```

```
</Para>
```

```
</Boek>
```

2.Structuur XML document

- Oefeningen

```
<?xml version="1.0" standalone="yes" ?>
```

```
<Para>Mijn eerste paragraaf.</Para>
```

```
<Para>Mijn laatste paragraaf</Para>
```

```
<?xml version="1.0" standalone="yes" ?>
```

```
<Boek>
```

```
<Titel>Professional <XML> van Wrox</Titel>
```

```
<Para>'Professional <XML>' van Wrox, het XML handboek!</Para>
```

```
</Boek>
```

```
<?xml version="1.0" standalone="yes" ?>
```

```
<Boek 1auteur="Léon">
```

```
  <Titel>Professional XML</Titel>
```

```
  <Para>Blabla</Para>
```

```
</Boek>
```

2. Structuur XML document

- Oefeningen
 - Welk van de 2 XML-documenten is correct?

```
<?xml version="1.0" standalone="yes" ?>  
<Boek isbn="123-4567-89">  
<Para>blabla</Para>  
</Boek>
```

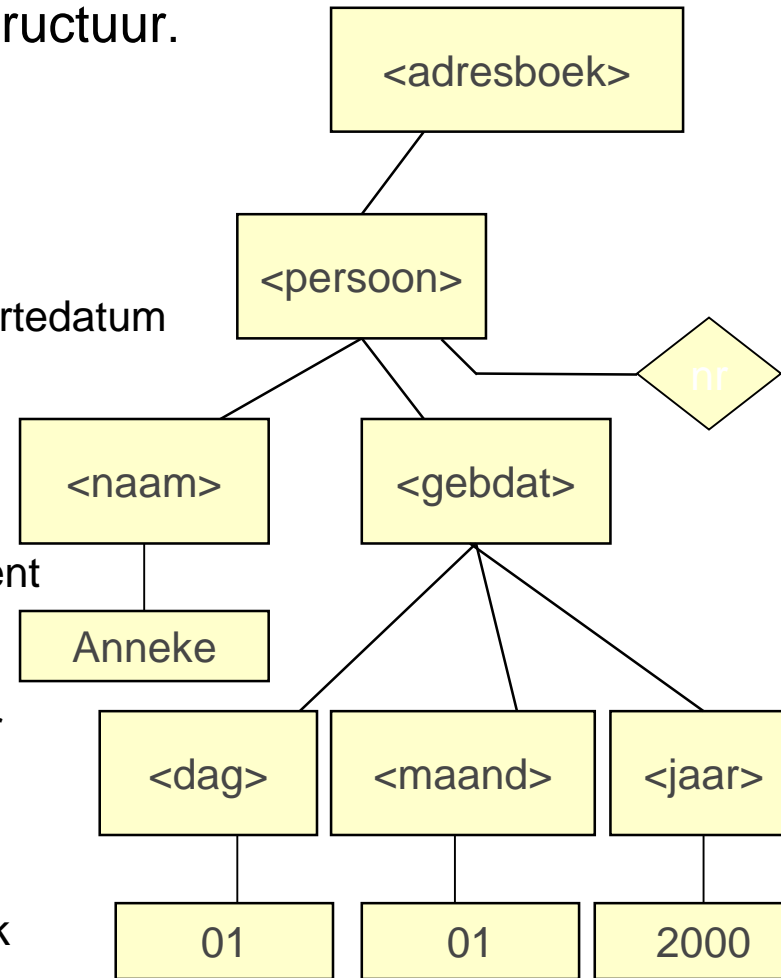
of

```
<?xml version="1.0" standalone="yes" ?>  
<Boek><Isbn>123-4567-89</Isbn>  
<Para>blabla</Para>  
</Boek>
```

2. Structuur XML document

e. XML Hiërarchische structuur (document tree)

- Well-formed XML-document = boomstructuur.
- Vereiste : 1 root.
- Tussen elementen bestaan relaties
 - Parent
 - Vb.adresboek is 'parent' van persoon;
persoon is parent van naam en geboortedatum
 - Child
 - Vb. naam en geboortedatum zijn 'children' van persoon
 - Sibling
 - Sibling elementen delen dezelfde parent
 - Vb. naam en geboortedatum
 - Ancestor
 - parent maar ook over-over-grootouder
 - Vb. adresboek is 'ancestor' van dag
 - Descendant
 - Kind ,... achter-achter-achterkleinkind
 - Vb. dag is 'descendant' van adresboek



3. Opbouwen van XML document

- Welke tags en bijbehorende attributen heb je nodig? Welke naamgevingen gebruik je?
- Hoe zal je de data structureren?
 - welke applicaties zullen van de data gebruik maken?
 - via welke toegangspaden?
 - hiërarchie?
 - moeten we data dupliceren? Wat met m:n?
 - Voorbeeld : Student - Vak

3. Opbouwen van XML document

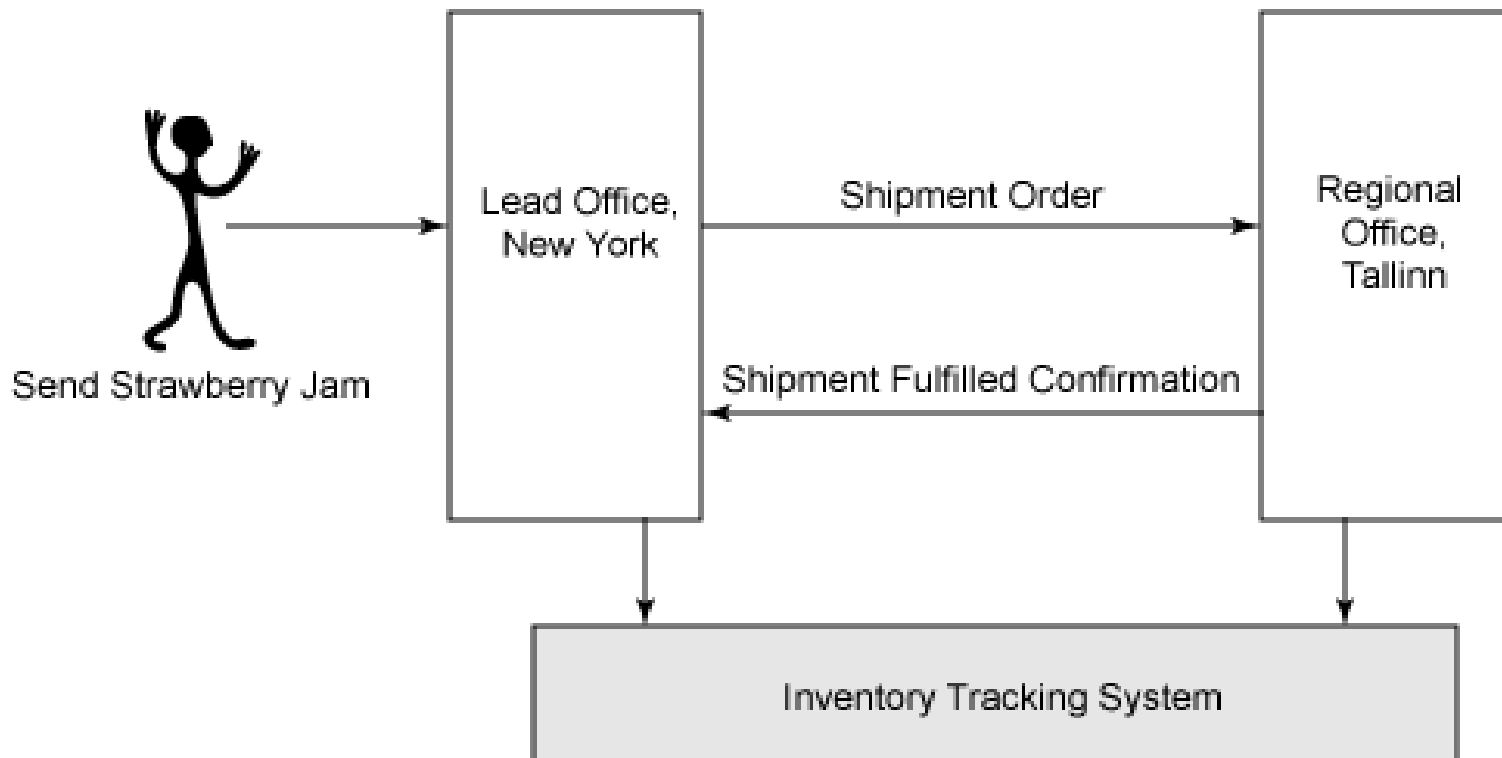
- Voorbeeld : Slechte opbouw van een XML-document :

```
<groente>Tomaat</groente>  
<prijs>1.49 Euro per Kg</prijs>  
<groente>Witloof</groente>  
<prijs>2.00 Euro per Kg</prijs>
```

- Wat als je de prijs wenst te weten van een product?
- Wat als je prijsberekeningen wenst te doen?

3. Opbouwen van XML document

- Vertrek van een UML klassendiagram
 - Voorbeeld: BALTIC Shipping workflow



3. Opbouwen van XML document

– Klassendiagram

BUSINESS CONCEPTS

UML DIAGRAM

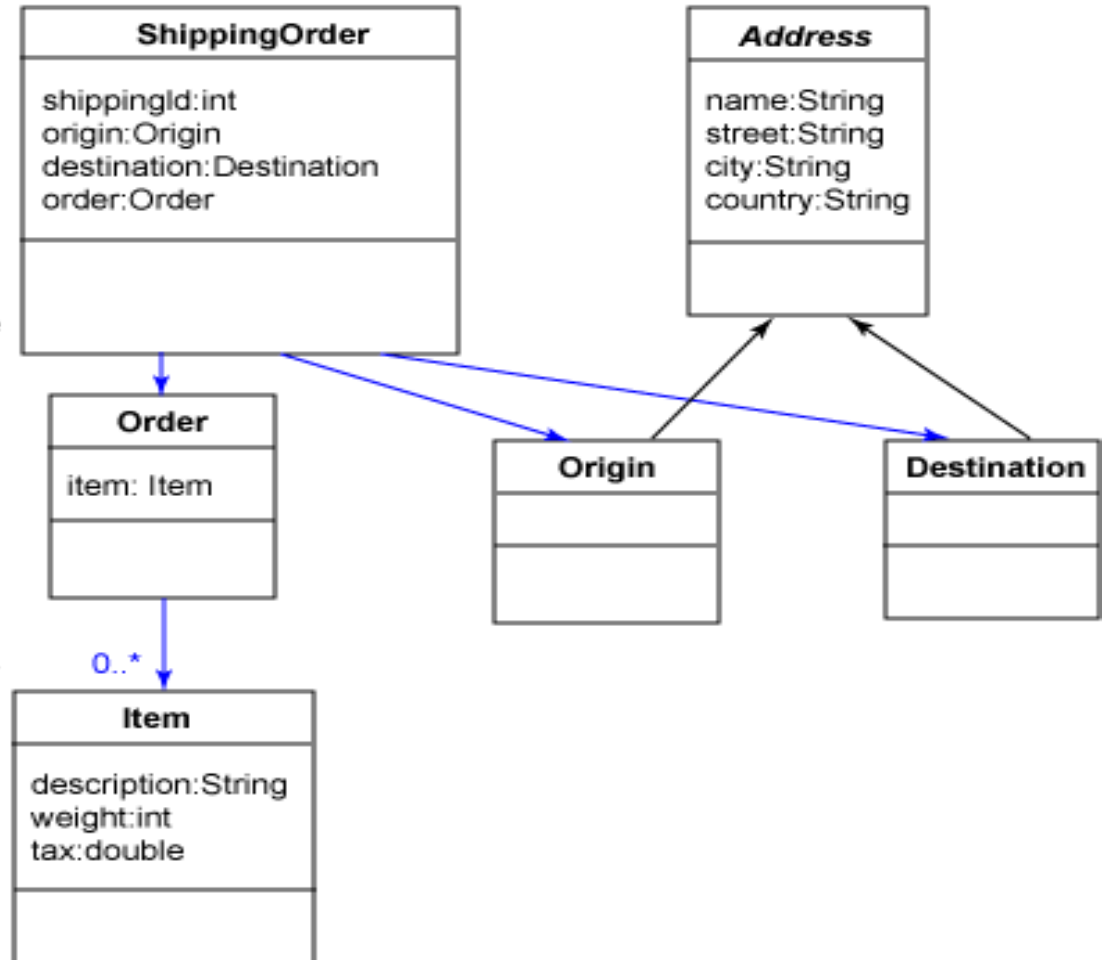
A Shipping Order has:
ShippingId
Origin
Destination
Order

Origin and Destination are
both of type Address

An Address has:
Name
Street
City
Country

An **Order** consists of one or
more Items

Each Item has
Description
Weight
Tax



3. Opbouwen van XML document

– ShippingOrder.xml

```
<?xml version="1.0"?>
<shippingOrder>
  <shippingId>09887</shippingId>
  <origin>
    <name>Ayesha Malik</name>
    <street>100 Wall Street</street>
    <city>New York</city>
    <country>USA</country>
  </origin>
  <destination>
    <name>Mai Madar</name>
    <street>Liivalaia 33</street>
    <city>Tallinn</city>
    <country>Estonia</country>
  </destination>
  <order>
    <item>
      <description>Ten Strawberry Jam bottles</description>
      <weight>3.141</weight>
      <tax>7.60</tax>
    </item>
  </order>
</shippingOrder>
```

4. Levenscyclus XML document

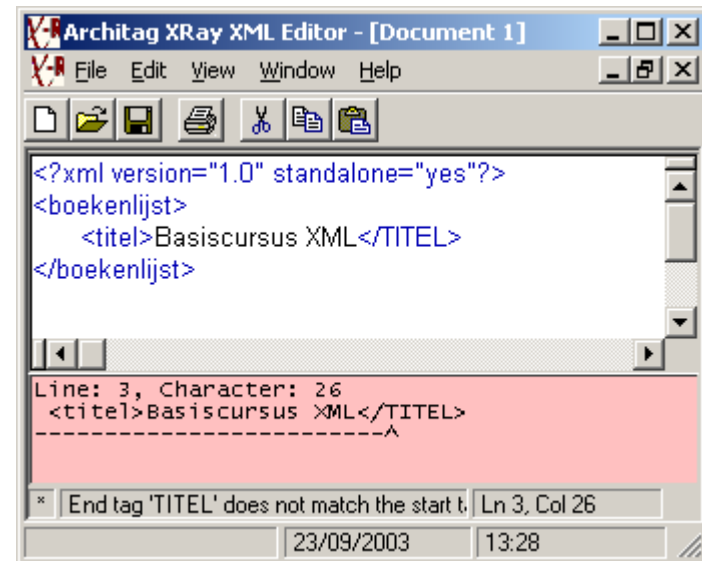
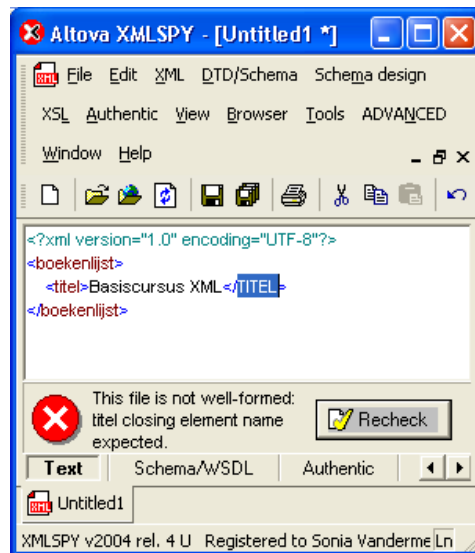
- Levenscyclus van XML document



4. Levenscyclus XML document

- Editors

- Oxygen : <http://www.oxygenxml.com/>
- Altova XML Spy : <http://www.xmlspy.com/>
- DataDirect Stylus Studio : <http://www.stylusstudio.com/>
- Eclipse
- Xray : <http://architag.com/xray/>
- Voor meer info : <http://www.xmlsoftware.com/editors.html>



4. Levenscyclus XML document

- XML Parser
 - Een software programma dat een XML document overbrengt naar het geheugen ter voorbereiding voor het manipuleren, ondervragen of transformeren van data.
 - Leest (sequentieel)
 - Interpreteert
 - 2 modi
 - » Non validating : Controleert syntax conform de regels van XML 1.0
 - » Validating : + controle op conformisme met een DTD of schema
 - Rapporteert fouten indien document niet well-formed/valid is (stopt bij eerste fout)
 - Doorgeeft aan een applicatie : De door de parser gegenereerde code kan vervolgens door de applicatie aangesproken worden via een API. De meeste parsers bieden hierbij ondersteuning voor DOM en/of SAX

4. Levenscyclus XML document

- XML Parser

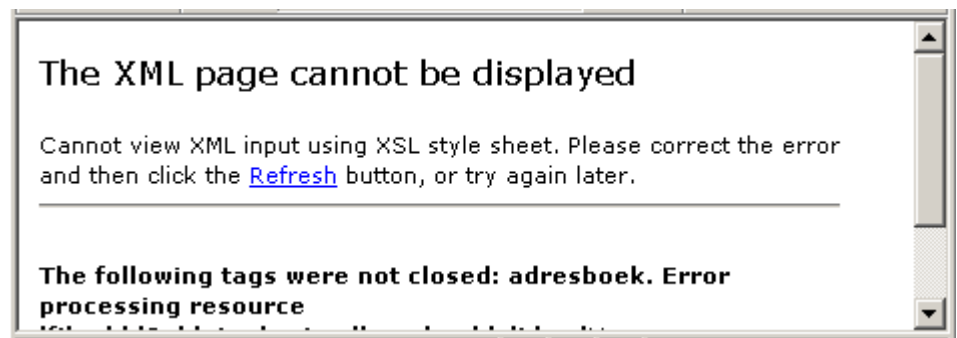
- Saxon (Michael Kay) (Java/.Net versie) :
<http://saxon.sourceforge.net>
- Apache Xerces (C en Java versie) :
<http://xml.apache.org/xerces2-j/index.html>
- Microsoft : MSXML 4.0/6.0 en .Net
<http://msdn.microsoft.com/downloads/webtechnology/xml/msxml.asp>
- Oracle XML parser for Java
http://technet.oracle.com/tech/xml/parser_java2/
- Intel XSS
- libXML
- AltovaXML (core van XMLSpy, open source)
- Voor meer info zie :
http://www.topxml.com/parsers/other_parsers.asp

4. Levenscyclus XML document

- Browsers en andere applicaties
 - Parser geeft boomstructuur door aan de clientapplicatie
 - Als applicatie een browser is dan worden de gegevens getoond aan de gebruiker. IE zal de gegevens opmaken
 - Voorbeeld Internet Explorer, gebruikt MSXML parser

Of igtv fout

```
<?xml version="1.0" ?>
<!-- Mijn adresboek -->
- <adresboek>
- <persoon nr="022566">
  <naam>Joske Vermeulen</naam>
  - <gebdat>
    <dag>19</dag>
    <maand>5</maand>
    <jaar>1986</jaar>
  </gebdat>
</persoon>
</adresboek>
```



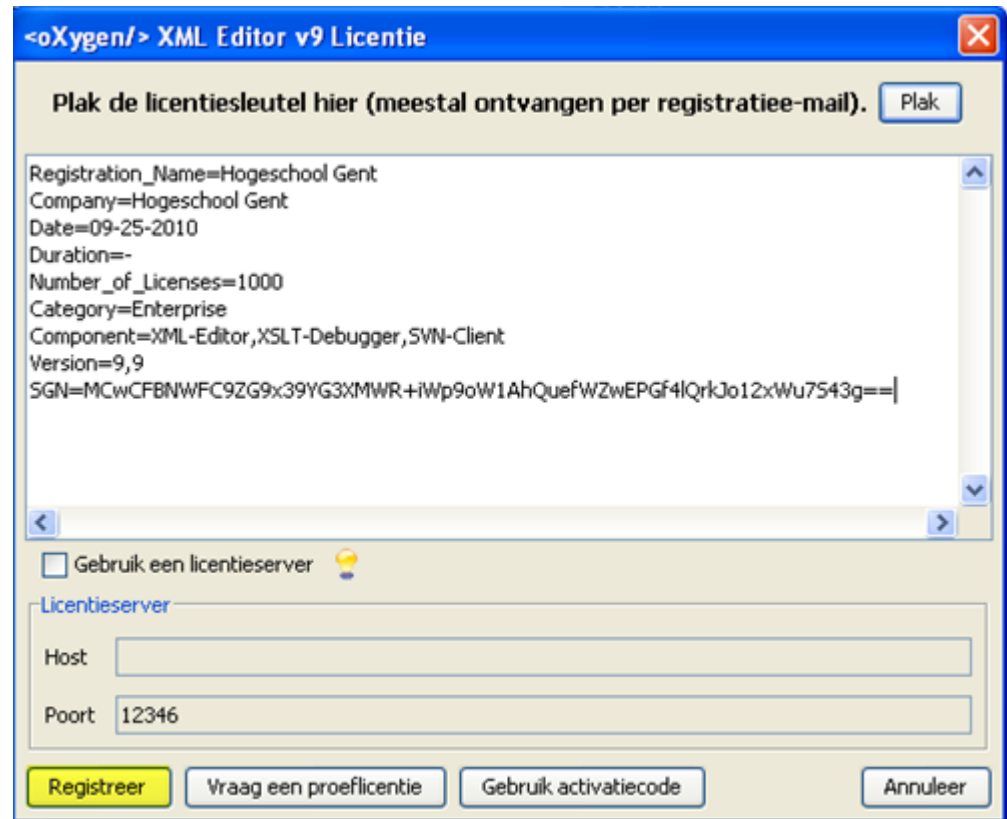
XML document bevat data, geen formattering
IE formatteert document mbv een stylesheet

On the Web en referenties

- <http://www.w3schools.com> » tutorial
- <http://www.w3.org/XML/>
» de officiële referentie van de XML specificatie, veel links naar software, tutorials, publicaties,...
- <http://www.xml.com/pub/a/axml/axmlintro.html>
» de xml specificatie, doorspekt met commentaren
- <http://xml.coverpages.org/>
» complete en up to date referentie-site, nieuwtjes, tools,... ivm SGML/XML en aanverwanten
- <http://xml.apache.org/>
» initiatief voor het bevorderen van XML gebruik, door aanbieden van kwaliteits software, gratis en open source
- SAI – avondconferentie : 11 jaar XML, successes and failures? Paul Hermans

Software: Oxygen

- Plak de volledige sleutel en klik op Registreer
- Oxygen start op.
- Klaar is Kees!



The screenshot shows the 'Oxygen XML Editor v9 Licentie' dialog box. It has a blue title bar with the text '<oXygen/> XML Editor v9 Licentie' and a close button. The main area is light yellow and contains the text 'Plak de licentiesleutel hier (meestal ontvangen per registratie-mail).' followed by a 'Plak' button. Below this is a text area containing the following information: Registration_Name=Hogeschool Gent, Company=Hogeschool Gent, Date=09-25-2010, Duration=-, Number_of_Licenses=1000, Category=Enterprise, Component=XML-Editor,XSLT-Debugger,SVN-Client, Version=9,9, and SGN=MCwCFBNWFC9ZG9x39YG3XMWR+iWp9oW1AhQueFWZwEPGf4lQrkJo12xWu7S43g==. Below the text area is a checkbox labeled 'Gebruik een licentieserver' with a lightbulb icon. Underneath the checkbox is a section titled 'Licentieserver' with two input fields: 'Host' and 'Poort' (containing '12346'). At the bottom are four buttons: 'Registreer' (highlighted in yellow), 'Vraag een proeflicentie', 'Gebruik activatiecode', and 'Annuleer'.

<oXygen/> XML Editor v9 Licentie

Plak de licentiesleutel hier (meestal ontvangen per registratie-mail). Plak

Registration_Name=Hogeschool Gent
Company=Hogeschool Gent
Date=09-25-2010
Duration=-
Number_of_Licenses=1000
Category=Enterprise
Component=XML-Editor,XSLT-Debugger,SVN-Client
Version=9,9
SGN=MCwCFBNWFC9ZG9x39YG3XMWR+iWp9oW1AhQueFWZwEPGf4lQrkJo12xWu7S43g==

☐ Gebruik een licentieserver

Licentieserver

Host

Poort 12346

Registreer Vraag een proeflicentie Gebruik activatiecode Annuleer

Software: Oxygen

Download de software van

http://www.oxygenxml.com/software_archive_editor.html

Onderstaande licentie is geldig voor studenten voor **versie 14.**

Plak alle 9 de regels in het registratiedialoogvenster dat je kan openen via het menu Help > Register (standalone)

of Windows > Preferences > oXygen > Register (Eclipse plugin)

-----START-LICENSE-KEY-----

Registration_Name=HoGent Bedrijf en Organisatie

Company=HoGent Bedrijf en Organisatie

Category=Academic-Departmental

Component=XML-Editor, XSLT-Debugger, Saxon-SA

Version=14

Number_of_Licenses=1

Date=12-05-2012

Maintenance=365

SGN=MC4CFQCVDHHDvxXvWyHrGo8wl/r6xtEdhwIVAIEDkNXhBaRZkEkOCP4W2n4IR6O

-----END-LICENSE-KEY-----

oXygen heeft intussen nieuwe versies uit, maar deze zal niet aangekocht worden.